Regression Testing of Web Services Using Parsing and Test case Prioritization Approach

Shaveta Sehgal
Desh Bhagat University,Mandi Gobindgarh

Er. Deepinderjeet kaur Dhaliwal
Department of Computer ScienceEngineering
Desh Bhagat University, Mandi Gobindgarh

Abstract- *Web services are the basic building blocks for every e-business applications now's- a-day. They provide efficient reusability mechanism, thereby reducing the development time and cost. Mostly the source code of web services is unavailable to other developers who use these services. Thus automated testing needs to be developed for testing these web services, which is possible by adding semantics to web services description language (WSDL). Also the test case reduction technique is very much required for regression testing. Previous work generates test cases for web services using case pri8oritization is need to done in reducing the test effort for web services using reducing techniques pair wise testing (PWT) and orthogonal array testing (OAT).in this work test case prioritization is need to done in reducing the testing effort for regression testing. To implement priority base testing than find test specification for web based data. After that perform orthogonal testing, pair wise testing and compare with strength generation.*

Keywords: *Web Services, Regression Testing, OAT, PWT.*

1. INTRODUCTION

1.1 Web Services: Web service is a method of communication that allows two software systems to exchange this data over the internet. The software system that requests data is called a service requester, whereas the software system that would process the request and provide the data is called a service provider. Different software might be built using different programming languages, and hence there is a need for a method of data exchange that doesn't depend upon a particular programming language. Most types of software can, however, interpret XML tags.

1.2 Types of Web Services

1.2.1 Big web services: Big web services are based on SOAP standard and often contain a WSDL to describe the interface that the web service offers. The details of the contract may include messages, operations, bindings, and the location of the web service. Big web services includes architecture to address complex non-functional requirements like transactions, security, addressing, trust, coordination, and also handles asynchronous processing and invocation.

1.2.2 REST FUL Web Services: RESTFUL web services are based on the way how our web works. Our very own world wide web (www) – the largest distributed application – is based on an architectural style called REST – Representational State Transfer. REST is neither a standard nor a protocol. It is just an architectural style like say for example client-server architecture (client-server is neither a standard nor a protocol). Web services following this architectural style are said to be REST ful Web services. So what is this REST? According to Roy Fielding who coined this term, "Representational State Transfer is intended to evoke an image of how a well-designed Web application behaves: Presented with a network of web pages (a virtual state-machine), the user progresses through an application by selecting links (state transitions), resulting in the next page (representing the next state of the application) being transferred to the user and rendered for their use."

1.3 Web Application: A web application or web app is any software that runs in a web browser. It is created in a browser-supported programming language (such as the combination of JavaScript, HTML and CSS) and relies on a web browser to render the application. Web applications are popular due to the ubiquity of web browsers, and the convenience of using a web browser as a client, sometimes called a thin client. The ability to update and maintain web applications without distributing and installing software on potentially thousands of client computers is a key reason for their popularity, as is the inherent support for cross-platform compatibility. Common web applications include webmail, online retail sales, online auctions, wikis and many other functions.

1.4 Testing: Software testing is an investigation conducted to provide stakeholders with information

ISSN: 2278 – 7798

about the quality of the product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation.

1.5 Regression Testing of Web service: Functional and non-functional Web service testing is done with the help of WSDL parsing and regression testing is performed by identifying the changes made thereafter. Web service regression testing needs can be categorized in three different ways, namely, changes in WSDL, changes in code, and selective re-testing of Web service operations. To capture the above, three intermediate forms of WSDL; namely, Difference WSDL (DWSDL), Unit WSDL (UWSDL), and Reduced WSDL (RWSDL), respectively can be used. These intermediate forms are then combined to form Combined WSDL (CWSDL) that is further used for regression testing of the Web service. This will help in Automatic Web Service Change Management (AWSCM), by performing the selection of the relevant test cases to construct a reduced test suite from the old test suite.

1.6 Types of Regression Testing:

1.6.1 Final Regression Tests: - A "final regression testing" is performed to validate the build that hasn't modified for some amount of time. This build is deployed or shipped to customers.

1.6.3 Regression Tests: - A standard regression testing is performed to verify if the build has NOT broken the other elements of the application by the recent code changes for defect fixing or for improvement.

2. RELATED WORK

McMaster, S. Et al. [1] "Driven Automated Testing Tool for Web Applications "Creating automated tests that exercise a web application through a browser is a challenging and time-consuming process. In this paper, we present a new tool currently under open-source development, Web Testing Explorer, which uses runtime state from the web application as feedback to search for defects in real-time and automatically create increasingly longer test cases with oracles for later execution while providing testers with the flexibility needed to deal with numerous web testing challenges.

Leotta, M **et al**.[2] "programmable web testing: An empirical assessment during test case evolution" There are several approaches for automated functional web testing and the choice among them depends on a number of factors, including the tools used for web testing and the costs associated with their adoption. In this paper, we present an empirical cost/benefit analysis of two different categories of automated functional web testing approaches: (1) capture-replay web testing (in particular, using Selenium IDE); and, (2) programmable web testing (using Selenium Web Driver). On a set of six web applications, we evaluated the costs of applying these testing approaches both when developing the initial test suites from scratch and when the test suites are maintained, upon the release of a new software version. Results indicate that, on the one hand, the development of the test suites is more expensive in terms of time required (between 32% and 112%) when the programmable web testing approach is adopted, but on the other hand, test suite maintenance is less expensive when this approach is used (with a saving between 16% and 51%). We found that, in the majority of the cases, after a small number of releases (from one to three), the cumulative cost of programmable web testing becomes lower than the cost involved with capture-replay web testing and the cost saving gets amplified over the successive releases.

Lei Xu et al [3] "A framework for Web applications testing"_Web application testing is concerned with numerous and complicated testing objects, methods and processes. So a testing framework fitting for the properties of Web application is needed to guide and organize all the testing tasks. Based on the analysis for Web application characters and traditional software testing process, the process for Web application testing is modelled, which describes a series of testing flows such as the testing requirement analysis, test cases generation and selection, testing execution, and testing results analysis and measurement. Furthermore, the realization techniques are also investigated so as to integrate each testing step and implement the whole testing process harmoniously and effectively. Thus the framework is suitable for the Internet environment and can guide the Web application testing actively and availably.

Boumiza, D. S. et al.[4] "Design and development of a user interface to customize web testing scenarios " A fact that we cannot deny is, web applications testing has become a very large scale, hence the interest of developers and especially testers to find the best solutions ever in order to improve web applications testing quality. Many web testing tools provide to user the "record and replay" feature that, like any other testing process, aims to compare actual results with expected ones to ensure the application

ISSN: 2278 – 7798

functionality. But this approach has failed functionally speaking despite its widespread in most of web testing tools. In fact, the challenge of most developers is to ensure quality by increasing the test accuracy and at the same time speeding up testing process. Facing those urgent needs, and these actual facts (record and replay approach) it became necessary to find another approach, from this came the idea to create model-based tests. Model-based testing is a new approach in web testing world and unlike record and replay technique, just a few tools relay on. Due to all its functional advantages, we have chosen to create a web testing tool that is based on this powerful technique and improve it, in a way that simplify modelling implementation in order to reduce the rate of errors.

Zhong sheng Qian et al. [5] "A Practical Web Testing Model for Web Application Testing" As an important method to ensure the quality of Web applications, Web testing attracts more and more attentions in the academic community and industrial world. Testing Web applications raises new problems and faces very high challenges. This work proposes a Web testing model for Web application testing. It starts from constructing the PFD (Page Flow Diagram) of the Web application. An algorithm is then designed to derive a PTT (Page Test Tree) from the PFD. From the PTT, a test translator is employed to extract the path expression, in order to generate all the test paths and then translates them into a test specification in XML syntax, which is the input of test engine. The test engine generates test cases and then executes them, and finally produces test report.

3. PROBLEM FORMULATION

Web services are the basic building blocks for every e-business applications now's- a-day. They provide efficient reusability mechanism, thereby reducing the development time and cost. Mostly the source code of web services is unavailable to other developers who use these services. The manual effort spent by them in testing these web services is very large in order to increase the interoperability. Thus automated testing needs to be developed for testing these web services, which is possible by adding semantics to web services description language (WSDL). Also the test case reduction technique is very much required for regression testing. Previous work generates test cases for web services using case pri8oritization is need to done in reducing the test effort for web services using reducing techniques pair wise testing (PWT) and orthogonal array testing (OAT).in this work test case prioritization is need to done in reducing the testing effort for regression testing. This work will consider test case prioritization technique for web services based on certain criteria and determine the better testing technique by considering the testing efficiency of each case.

4. RESULTS

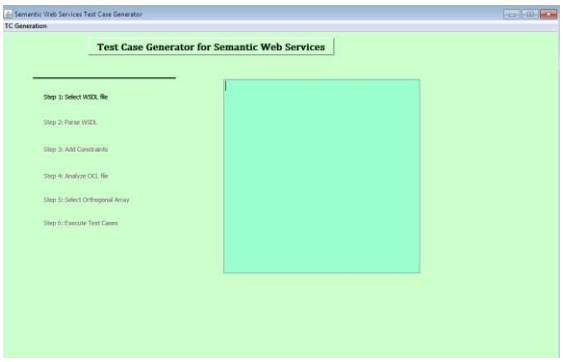

Fig 5.1: Fig 1 Loading of WSDL

In this first GUI, the first step is to load the WSDL (Web Service description language) file. If the file is a valid WSDL then we can move onto the next step.

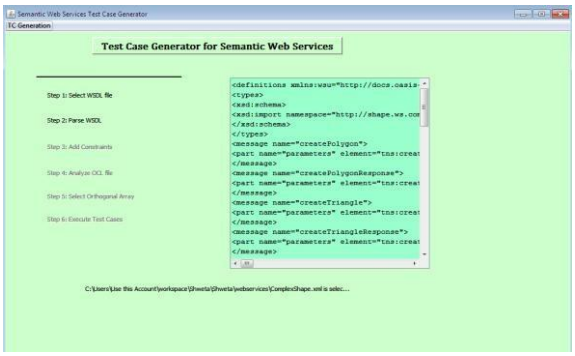

Fig 5.2: Represent Component Extraction

In the second step loaded WSDL file is parsed and components are extracted.

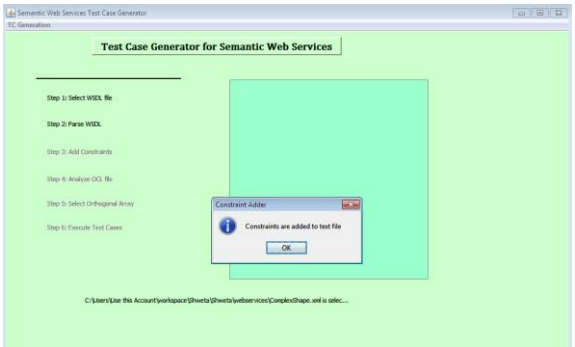

Fig 5.3**:** Represents Addition of Constraints to file

ISSN: 2278 – 7798

In third step constraints are added to the file.

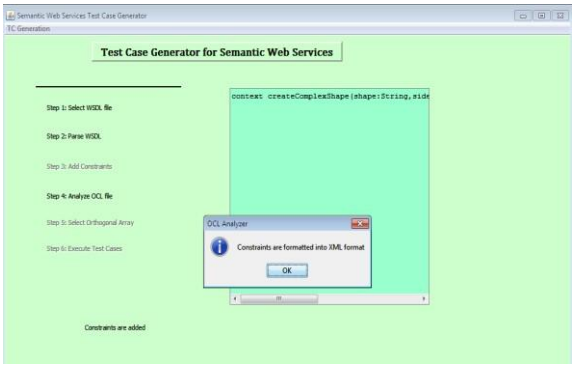

Fig 5.4: Conversion into XML format

Added constraints are formatted into XML format and are displayed in the text area

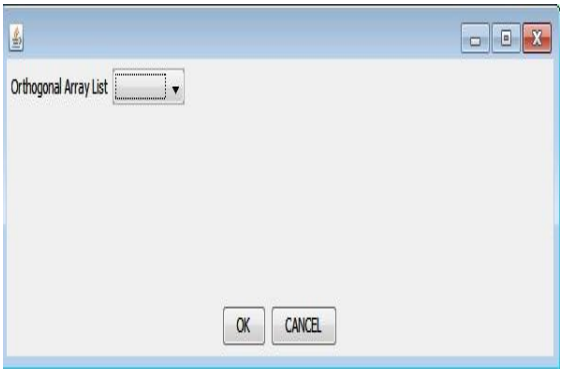

Fig 5.5: Represents Orthogonal Array List

Orthogonal Array List method is used to fill the input values and boundary of testing scenario.

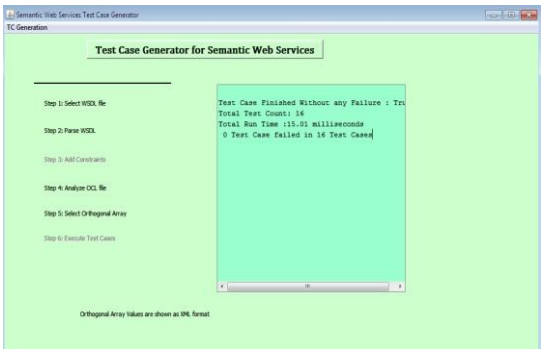

Fig 5.6: Execution of test cases

Test Cases are executed and results are displayed in the test area.

5. CONCLUSION

Web services the component that can be used in various applications of computers. In this process, these web services have been designed by using different descriptive languages. In the proposed work, different web services have been designed by using WSDL (web service descriptive language). This language is used to explain the web services that can be created and publish of the hosting server and can be used by different users. In this purposed work the semantic analysis has been done for the different web services. In the semantic analysis the web services source code has been divided into different syntaxes and on these syntaxes OCL descriptor is implemented that utilize different combinations. After the OCL the test cases have been generated for different web services and on the basis of these web services. These test cases have been implemented on these different web services and the executed for checking of validity of these different services. In this work three different types of web services have been used that have been parse by using parsing technique and after this these services can be used for execution of test cases. We got various types of parameters & on the basis of these parameters we conclude that our system gives us better results.

REFRENCES

[1] Siripol Noikajana, Taratip Suwannasart, "An Improved Test Case Generation Method for Web Service Testing from WSDL-S and OCL with Pair-wise Testing Technique", *in Proceedings of IEEE International Computer Software and Applications Conference*, 2009.

[2] Xiaoying Bai, Wenli Dong, W.-T. Tsai, and Y. Chen, "WSDL based automatic test case generation for Web services testing", *in Proceedings of IEEE International Workshop on Service-Oriented System Engineering*, 2005, pp.207-212.

[3] Chunyan Ma, Chenglie Du, Tao Zhang1, Fei Hu, "WSDL-Based Automated Test Data Generation for Web Service*", International Conference on Computer Science and Software Engineering*, vol. 2, pp. 731-737, 2008.

[4] John T. E. Timm, Gerald C. Gannod, "Specifying Semantic Web Service Compositions using UML and OCL", *in Proceedings of the IEEE International Conference on Web Services*, pp. 521-528, 2007.

[5] Sebastien Salva, Isaam Rabhi, "Automatic Web Services Robustness Testing from WSDL Descriptions", 2009,

[6] J Offutt, W. Xu, "Generating Test Cases for Web Services using Data Perturbation", *ACM sigsoft, Software Eng. Notes*, Vol. 29 (5), pp. 1-10, Sep. 2004.

[7] M. Hong and Z. Lu, "A framework for testing Web services and its supporting tool," *in Proceedings of the*

ISSN: 2278 – 7798

IEEE International Workshop Service-Oriented System Engineering, 2005, pp. 199- 206.

[8] Xiaoying Bai and Shufang Lee Wei-Tek Tsai and Yinong Chen, "Ontology-Based Test Modeling and Partition Testing of Web Services", *international conference on Web Services*, pp. 465-472, 2008.

[9] Sheng Gao "The measurement of Geospatial Web Service quality in SDIs", International Conf. on Geoinformatics, 2009, pp 1 – 6.

[10] Akingbesote, A.O. "A quality of service aware multi-level strategy for selection of optimal web service" IEEE Conf. on Adaptive Science and Technology (ICAST), 2013, pp 1 – 5

[11] Maolin Tang "A hybrid genetic algorithm for the optimal constrained web service selection problem in web service composition" IEEE Conf. on Evolutionary Computation (CEC), 2010, pp 1 – 8.

[12] Chao Ma "An Approach for Visualization and Formalization of Web Service Composition" IEEE Conf. on Web Information Systems and Mining, 2009, pp 342 – 346.

ISSN: 2278 – 7798